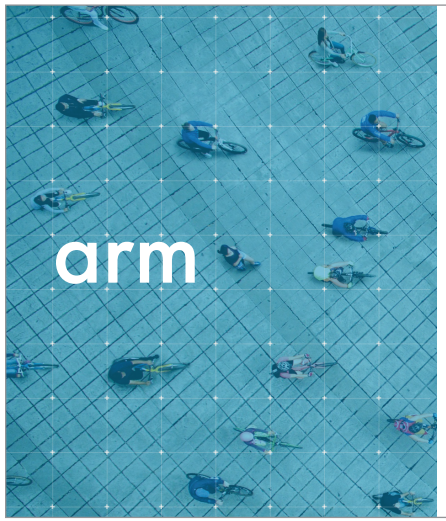
Achieving optimal PPA for Cortex®-A75 Arm® Processors using TSMC FinFET Technology with Cadence Digital Tools

Cadence Design Systems / Arm

ABSTRACT

The latest Cortex-A75 Arm CPU supporting Armv8-A architecture have been announced, which extend performance, while maintaining the power-efficient focus of Arm processors, to drive the innovations in high-end computing, mobile, and consumer markets. This learning session will highlight design implementation tips and tricks for a high performance, convergent and predictable RTL2GDS flow. Techniques such as physical aware synthesis, early clock estimation, layer aware optimization, signoff accurate, physical and IR Drop aware design closure, will be discussed. As the latest Arn processors increase in size, turn around time during implementation is becoming a concern. During this session, multi-thread and distributed computing technology will be reviewed, showing how run-time can be managed for large, highly complex designs. Each implementation of a Arm core has something unique from the designers' viewpoint, so the end goal of this session is to leave the designer with clear guidance on how to achieve the optimal PPA by providing the best starting point for efficient leakage and dynamic power optimization, highest frequency, or ability to tradeoff Performance vs Power/Area.



Achieving optimal PPA for Cortex®-A75 Arm® Processors using TSMC FinFET Technology with Cadence Digital Tools

Himanshu Chopra, Design Engineer, ARM
Vidit Babbar, Principal Engineer, ARM
Paddy Mamtora, Product Engineering Group Director, Cadence

TSMC OIP, San Jose
13th September 2017

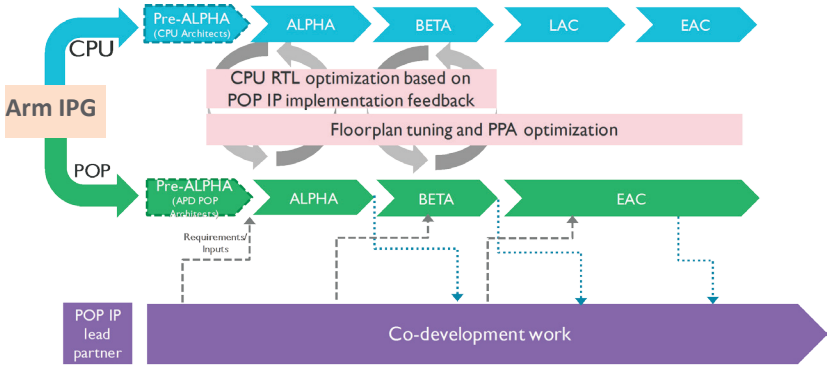
Agenda

- Arm POP™ IP - Introduction & Context
- High-Performance Cadence® Genus™ Synthesis/Innovus™ Implementation/Tempus™ Timing CPU Flow
- Synthesis flow tuning for faster TAT
- Optimal Placement to jump over timing bottlenecks
- Clock Tree – Structural choices; ECF & FlexHtree
- Crosstalk Closure - Issues & Techniques
- SignoffOptDesign to push the performance
- Recovering the last "mW" for LITTLE cores
- Summary & Key takeaways

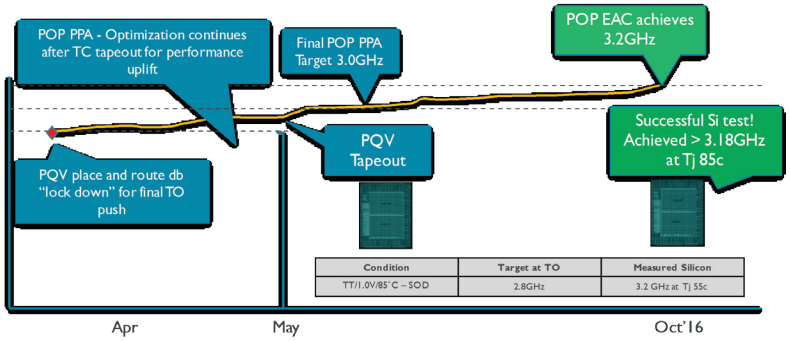
Arm POP IP on TSMC 16nm

Need to shorten the design cycle	POP IP is a comprehensive, fully validated Cortex-A CPU implementation solution Includes Physical IP, floorplans and reference implementation scripts
Need to lower technical and schedule risk	POP IP is developed and tuned in synergy with RTL over several iterations All Physical IP and implementation issues have been identified and solved by EAC date
Need to achieve market-leading PPA	POP undergoes extensive iterative floorplan exploration and design tuning to deliver market- leading PPA Our record in 16nm FinFET technology is a testament to the hard work behind POP development

PDG collaboration with processor design on POP IP



Cortex-A73 PQV and POP IP Development Cycle



5

arm

Cadence High-Performance CPU Flow

- Extreme Flow for CPU designs
- Full Clock aware place / opt
- High timing effort placement
- Layer aware opt
- Power activity driven placement

- Timing driven routing
- Track aware opt & SI avoidance

- PBA based Opt
- Setup opt with advanced clock skew
- Dynamic and Leakage power
- Area Recovery

POWER OPT

Innovus Implementation

Genus Synthesis (physical)
place_opt_design
ccopt_design
optDesign –postCTS [-hold]
routeDesign
optDesign –postRoute –setup [–hold]
signoffOptDesign PBA based Opt (Tempus ECO)
Tempus Signoff STA

USEFUL SKEW

- Physical aware generic and mapping
- Full physical → guidance to Innovus

- Clock Concurrent Optimization
- CTS + useful skew + post-CTS opt
- FlexHtree / Mesh + useful skew
- Low power clock flow
- Auto-selection of clock cells

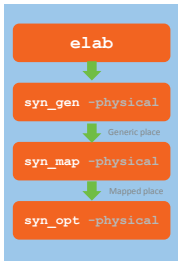
- PostRoute Opt
- Useful Skew based Opt
- Wire/Via opt

6

arm

Genus Synthesis flow tuning for faster TAT

Genus Synthesis



High capacity enables synthesis of whole CPU flat

Shares same engines with Innovus Implementation which allows tight correlation between synthesis and P&R

Floorplan incorporates the power grid and power gates from synthesis

Few modules are marked as "DON'T UNGROUP" for synthesis

Several Path Groups defined for better critical path optimization

7

arm

signoffOptDesign / Tempus ECO Signoff PBA-based optimization

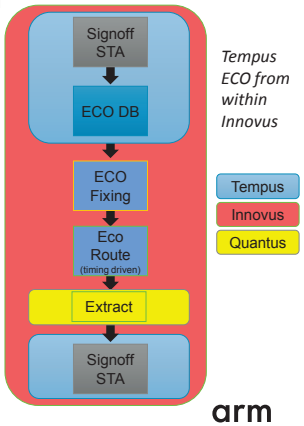
Signoff PBA-based optimization
- Setup, hold, leakage, dynamic, area, clock skew

Fully integrated within Innovus Implementation

Supports Distributed-MMMC

Differentiation – Setup timing opt
- Setup opt using Multi-level clock skew
- Optimizes cell, wire and SI delays
- Fully layout aware ECOs

Differentiation – Power opt
- Concurrent leakage and dynamic power opt
- Vector based dynamic power opt



8

arm

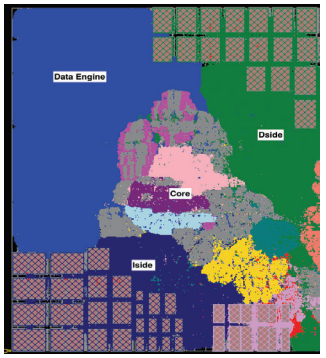
Optimal Placement to jump over timing bottlenecks

- Placement is a key focus area for high performance Arm core implementations
  - Sub-optimal combinations can result in crosstalk sensitive zones, local utilization hotspots OR Congested areas that lead to DRCs
- Two-phased approach is needed for next-generation Arm cores:
  - High-level data flow driven placement
    - Involves macro floorplanning which dictates the positioning of main modules
  - Unit level placement analysis
    - Break the high-level functional modules into granular blocks
    - Analyze the timing & placement for these in conjunction with data flow requirements
- Ensures a scalable floorplan/placement which does not saturate due to utilization/crosstalk issues

9

arm

Cortex-A75 High-level data flow driven placement

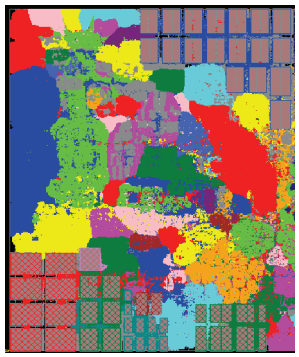


- Identification of Major logic modules
  - Ifetch; Idecode; Dispath; Loadstore etc.
  - Optimal Placement location for each logic module
- Module Placement derived from Data flow
  - Place Iside closer to Instruction Cache
  - Dside should sit closer to Data cache
  - Data engine should be moved towards the top-left
- No bounds/regions needed with Innovus Implementation
  - Ideal to have the placer guide the modules in accordance with data flow

10

arm

Cortex-A75 Unit level placement analysis for scalable performance



- High-level module placement doesn't ensure performance scalability
- UNIT level placement analysis is a MUST
  - Identify and classify unit level modules as:
    - Interaction dominant modules
    - Timing critical modules from a logic depth perspective
    - Decoupled modules – Sit far from I/O & Memories
  - Correlate the placement of unit level modules with data flow and analyze timing criticality
  - Performance Scalability concerns:
    - Heavily split / scattered unit modules (undesired)
    - Misplaced unit modules (w.r.t data flow)
    - Unit modules elongated due to macro or I/O dependencies
- Arm POP IP encompasses floorplans which can scale performance beyond 3.2GHz (16nm)

11

arm

Clock Tree: Structural choices to tune PPA

- Arm POP libraries support a wide range of INV/BUF variants
- Using N-type (balanced) INV are recommended for high performance Arm core implementations
  - Map this to the fastest Vt/CL to ensure minimal latency & OCV impact
- Mapping to N-type (balanced) BUF can improve performance & reduce Pdyn
  - Higher variance across corners w.r.t timing / duty-cycle
  - CCOPT skewing does see a benefit with BUF cells

- Optimal Path: N-type INV for CTS & N-type BUF for leaf level skewing

```
set_ccopt_property use_inverters  true
set_ccopt_property inverter_cells <list of trunk inverters>
set_ccopt_property buffer_cells {list of buffer cells in case there are don't_touch buffers in the trunk}
set_ccopt_property leaf_inverter_cells {}  # empty list
set_ccopt_property leaf_buffer_cells {list of leaf buffers}
```

Clock Tree Cell Choice (16FFC/11LM)	Performance @ TT-1v-85c	Clock Dynamic Power @ TT-0.8v-85c
N-type INV	1X	1X
F-type INV	0.95X	1.03X
R-type INV	0.98X	1.02X
N-type INV+BUF	1X	0.9X
N-type BUF	1.03X	0.85X

12

arm

Cortex-A75 Clock aware placement & optimization

- Early Clock Flow(ECF) for high performance CPUs
- Routing resource and congestion estimation at place / Opt factors clock nets accurately
- Correct layers gets picked for CTS nets routing estimation during place
- Placement / Opt factors CTS cells placement – Better timing convergence through the flow
- Better prediction of real critical paths to optimize
- Better modeling of Clock Gate latencies and optimization of Useful skew based on that

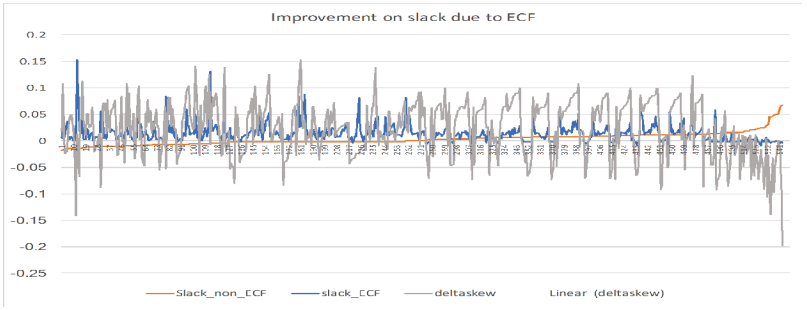
Postroute Timing	Without Full Clock Aware	Full Clock Aware
WNS (Reg2CG)	-0.042	-0.036
TNS (Reg2CG)	-3.933	-1.375
WNS (Reg2Reg)	-0.064	-0.067
TNS (Reg2Reg)	-77.383	-62.741

TNS reduced by 20%

13

arm

Cortex-A75 Clock aware placement & optimization



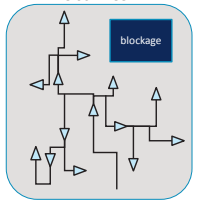
Improvement in slack with early clock flow for top-500+ paths

14

arm

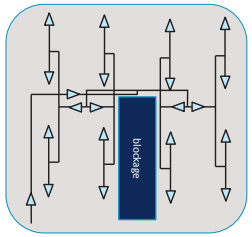
High Performance Clocking - FlexHtree / Clock Mesh

Clock Tree



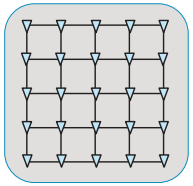
- Flexible placement
- Clock gating for power reduction
- Useful skew for timing closure
- Large latency depending on tree depth
- OCV can be significant
- High variability across corners

Flexible H-tree



- Flexible placement
- Clock gating at lower levels
- Some useful skew for timing closure
- Less OCV
- Less variability across corners

Clock Mesh

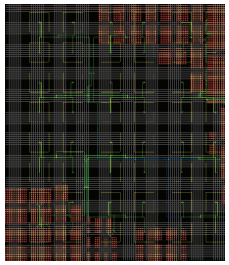


- Predefined placement
- Multi driven mesh wires
- Low skew and latency
- High power as whole mesh switching
- No OCV
- Little variability

15

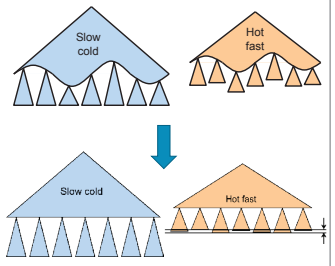
arm

Cortex-A75 FlexHtree Results



Final STA timing	BASELINE		Flex-Htree
setup reg2reg	WNS	-0.017	-0.011
	TNS	-1.434	-0.224
	FEPS	293	63
setup reg2cgate	WNS	-0.017	-0.011
	TNS	-0.024	-0.011
	FEPS	7	1

50-100MHz improvement using Flex-Htree

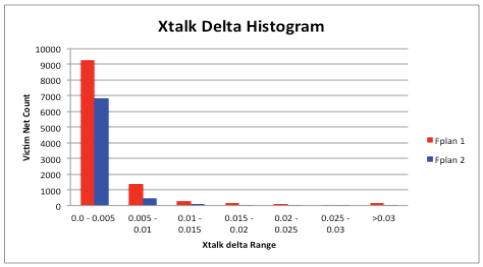


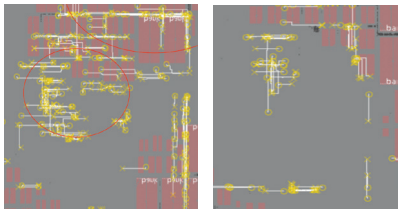
Better performance, along with other benefits of structurally balanced tree (better variability scaling etc.)

16

arm

Crosstalk : A bigger bottleneck than ever before!





Crosstalk effects play a major role in determining maximum performance of Arm cores on FinFET nodes

Crucial to reduce the clock paths cumulative crosstalk to <2-3ps

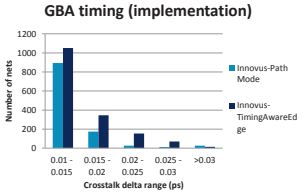
- Data net crosstalk highly dependent on module placement
- Analyze the crosstalk early in the design cycle to choose an optimal floorplan & placement strategy

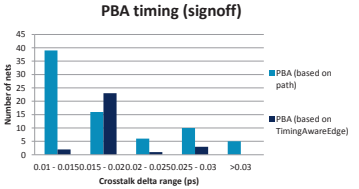
arm

Crosstalk Closure : Drive Correlation

A. Drive correlation:

- Use signoff extraction during last round of incremental post route
  - Enables identification of crosstalk bottlenecks for optimization
- Attacker alignment : timing_aware_edge
  - Removes pessimism from GBA
  - Enhanced correlation from implementation to signoff in PBA
  - Minor utilization bump in postRoute compared to 'path' mode of aggressor alignment





arm

Crosstalk Closure : Module Padding

B. Module padding on sensitive modules:

- Identification of crosstalk sensitive modules based on overall delta zone profiling
- Module padding helps in spacing out standard cells which can reduce local pin density & congestion
  - Effectively reduces crosstalk in critical modules
- Implemented as part of Innovus P&R flow:
  - setPlaceMode -modulePadding {u_cpu/u_I?/u_pipe} 1.05
  - setPlaceMode -modulePadding {u_cpu/u_dside/u_arb} 1.05
  - setPlaceMode -modulePadding {u_cpu/u_dside/u_iq} 1.10
- # of nets with crosstalk delta > 10ps reduces by 30%
  - Overall TNS reduction of 20%
- Scalable solution which can be used for Arm core implementations

arm

Crosstalk Closure : Layer Assignment

C. Layer assignment of hard macro interconnects

- A set of critical victim nets associated with areas between macro intensive channels
- Alternate V/H layers used for signal routing within ram channels → minimal timing-window and interconnect length overlap.
- Careful analysis of congestion in macro channels

D. Additional track spacing on critical nets

- Identify the collection of crosstalk sensitive nets and apply a preferred track spacing as a 'soft constraint'
- Extremely effective in minimizing crosstalk for interface nets at cluster level implementations

arm

Cortex-A75 signoffOptDesign / Tempus ECO Results Signoff PBA-based optimization

125 MHz improvement on >3GHz design

Techniques used

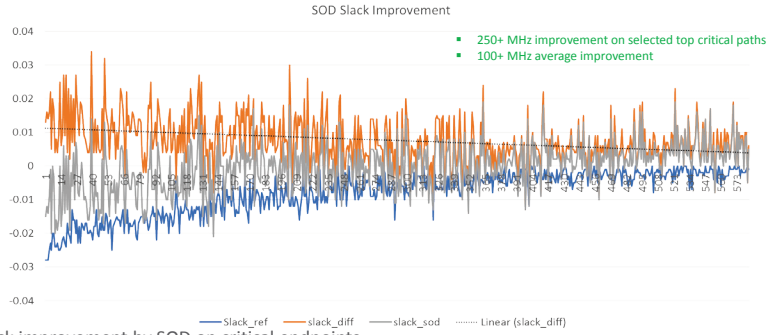
- Advanced clock skewing
- Data path delay fixing
- On-route buffering
- Routing aware cell sizing
- Timing driven ECO Route

	Before SOD			After SOD		
Path Group	WNS (ps)	TNS (ns)	FEP	WNS (ps)	TNS(ns)	FEP
REG2REG	-18	-0.853	168	-5	-0.255	64
RE2CLKGATE	-17	-0.064	16	-5	-0.024	17
REG2MEM	-8	-0.069	16	-2	-0.007	6
MEM2REG	-8	-0.16	49	-2	-0.004	3

21

arm

Cortex-A75 signoffOptDesign / Tempus ECO Results Signoff PBA-based optimization

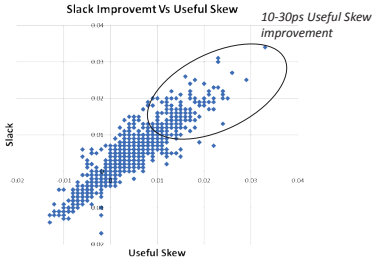


Slack improvement by SOD on critical endpoints

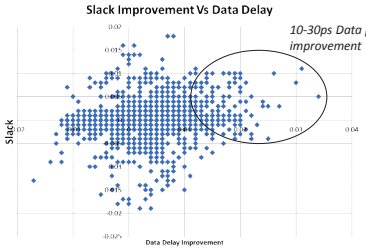
22

arm

Cortex-A75 signoffOptDesign / Tempus ECO Results Signoff PBA-based optimization



- Slack improvement through useful skew in SOD



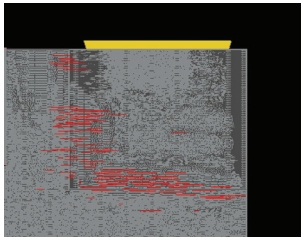
- Slack improvement by data path delay reduction

23

arm

Efficient Level Shifting across voltage domains

- Voltage domain crossings can become intensive in terms of interactions
  - Congestion sensitive on the boundary
- Insertion of multi-bit level shifters eases congestion
  - # of level shifters cells reduced by 75%
  - Area savings of 55% over single bit variants
- Native support in Genus Synthesis / Innovus Implementation
  - set_attribute use_multibit_cells true
  - merge_to_mulbit_cells [-logical] <design>

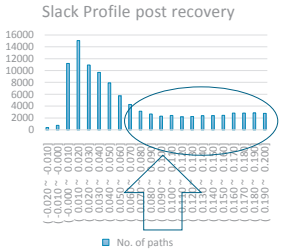


24

arm

Recovering the last “mW” for LITTLE cores

- Recipe for “LITTLE” Arm Core implementations focused on Leakage efficiency(DMIPS/mW):
  - Tuning the leakage flow iterations for Arm POP libraries
    - 12 Vt & Channel length flavors delivered as part of the POP
    - Important to identify the flavors which need to specified for maximum benefit
    - (A) Two pass leakage recovery : SVt OFF followed by SVt ON
    - (B) Two pass setup recovery : ULVt OFF followed by ULVt ON
  - Important to analyze the slack & transition profiles post recovery
    - Focus on I/O, DFT and RST paths
    - Make sure all the positive slack paths are already using slowest Vt/CL for combinational logic
  - “set_eco_opt_mode –pba_effort high” for aggressive leakage recovery
  - “set_eco_opt_mode –setup_recovery true” for keeping setup timing intact
  - “set_eco_opt_mode –allow_multiple_incremental true” enables recursive improvement



Positive slack zone needs detailed analysis to ensure optimal recovery

arm

Summary

- Driving the high performance Arm CPU implementations using POP IP offering with Cadence Implementation flow
  - Optimized Arm physical IP comprising standard cells and memory instances
  - TSMC 16nm FinFET technology enables additional gains on performance & power
  - Reference Flow Methodology for PPA closure on next generation Arm CPUs
    - Synthesis → STA tuning to achieve highest performance in a given power budget
    - High-level & UNIT level placement/floorplan recommendations
- Arm is using all the latest technologies from Cadence tool suite comprising Genus/Innovus/Tempus for PPA flow development
  - Early Clock Flow, FlexHTree and signoffOptDesign are the key technologies for pushing the performance envelope on next generation Arm CPUs

arm

arm

NOTE